UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/821,586	04/09/2004	Albert B. Barabas	11811-008002	1971

26161 7590 04/19/2007
FISH & RICHARDSON PC
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER	
TO, BAOQUOC N	
ART UNIT	PAPER NUMBER
2162	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/19/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/821,586	BARABAS ET AL.	
	Examiner	Art Unit	
	Baoquoc N. To	2162	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 January 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-83* is/are pending in the application.

4a) Of the above claim(s) *62-67* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-61 and 68-83* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *09 April 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *08/31/2005*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-83 are pending in this application.

Election/Restrictions

2. Applicant's election of Group I consisting claims 1-61 and 68-83 without traverse in the reply filed on 01/23/2007. Applicants are reminded to cancel the Non-Elected claims in the next communication.

Claims 1-61 and 68-83 are presented for examination.

Priority

3. This applicant is a continuation of and claims the benefit of priority form U.S. applicant Serial No. 09/687,941, filed October 13,2000.

Oath/Declaration

4. Oath/Declaration is corrected and accepted by the examiner.

Abstract

5. Abstract is in narrative form and generally limited to a single paragraph on a separate sheet within the range of 50 to 150 words.

Specification

6. The disclosure is objected to because of the following informalities: The arrangement of the specification is not in the required order. Appropriate correction is required.

The following guidelines illustrate the preferred layout for the specification of a utility application. These guidelines are suggested for the applicant's use.

Arrangement of the Specification

As provided in 37 CFR 1.77(b), the specification of a utility application should include the following sections in order. Each of the lettered items should appear in upper case, without underlining or bold type, as a section heading. If no text follows the section heading, the phrase "Not Applicable" should follow the section heading:

(a) TITLE OF THE INVENTION.
(b) CROSS-REFERENCE TO RELATED APPLICATIONS.
(c) STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT.
(d) THE NAMES OF THE PARTIES TO A JOINT RESEARCH AGREEMENT.
(e) INCORPORATION-BY-REFERENCE OF MATERIAL SUBMITTED ON A COMPACT DISC.
(f) BACKGROUND OF THE INVENTION.
 (1) Field of the Invention.
 (2) Description of Related Art including information disclosed under 37 CFR 1.97 and 1.98.
(g) BRIEF SUMMARY OF THE INVENTION.
(h) BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING(S).
(i) DETAILED DESCRIPTION OF THE INVENTION.
(j) CLAIM OR CLAIMS (commencing on a separate sheet).
(k) ABSTRACT OF THE DISCLOSURE (commencing on a separate sheet).
(l) SEQUENCE LISTING (See MPEP § 2424 and 37 CFR 1.821-1.825. A "Sequence Listing" is required on paper if the application discloses a nucleotide or amino acid sequence as defined in 37 CFR 1.821(a) and if the required "Sequence Listing" is not submitted as an electronic document on compact disc).

Drawings

7. Drawing filed on 04/09/2004 is corrected and accepted by the examiner.

Information Disclosure Statement

8. The information disclosure statement (IDS) submitted on 08/31/2005. The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

MPEP 2106 IV. B.2. (b)

A claim that requires one or more acts to be performed defines a process. However, not all processes are statutory under 35 U.S.C. 101. Schrader, 22 F.3d at 296, 30 USPQ2d at 1460. To be statutory, a claimed computer-related process must either: (A) result in a physical transformation outside the computer for which a practical application in the technological arts is either disclosed in the specification or would have been known to a skilled artisan, or (B) be limited to a practical application within the technological arts.

9. Claims 57-61 in view of the above cited MPEP section, are not statutory because claims they merely recite computing steps without producing any concrete and useful result and/or being limited to a practical application within the technological arts. Claim 57 lacks the necessary physical articles or objects to constitute a machine or a manufacture within the meaning of 35 USC 101. They are clearly not a series of steps

or acts to be a process nor are they a combination of chemical compounds to be a composition of matter. As such, they fail to fall within a statutory category. They are, at best, functional descriptive material *per se*. Descriptive material can be characterized as either "functional descriptive material" or "nonfunctional descriptive material." Both types of "descriptive material" are nonstatutory when claimed as descriptive material *per se*, 33 F.3d at 1360, 31 USPQ2d at 1759. When functional descriptive material is recorded on some computer-readable medium, it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized. Compare *In re Lowry*, 32 F.3d 1579, 1583-84, 32 USPQ2d 1031, 1035 (Fed. Cir. 1994). Merely claiming nonfunctional descriptive material, i.e., abstract ideas, stored on a computer-readable medium, in a computer, or on an electromagnetic carrier signal, does not make it statutory. See *Diehr*, 450 U.S. at 185-86, 209 USPQ at 8 (noting that the claims for an algorithm in *Benson* were unpatentable as abstract ideas because "[t]he sole practical application of the algorithm was in connection with the programming of a general purpose computer.")

Claim 58-61 are rejected under the same deficiency as claim 1.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-27, 32-43 and 48-52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hapner et al. (US. Patent No. 5,727,203) in view of Somani et al. (US. Patent No. 5,524,212).

Regarding on claim 1, Hapner teaches a method comprising:

maintaining a database that stores data persistently (col. 7, lines 15-16),

accepting tasks from the task sources, at least some of the tasks having competing requirements (multiple threads competing) for user of regions of the database (resources), each of the regions including data that is all either lock or not locked (commit) for writing at a given time (col. 9, lines 54-65).

Hapner does not explicitly teach associating each of the region with an available processor and distributing the jobs for concurrent execution by the associated processors; and defining, for each of the tasks, jobs each of which requires write access to regions that are to be accessed by no more than one of the processors. Somani teaches associating each of the region with an available processor and distributing the jobs for concurrent execution by the associated processors; and defining, for each of the tasks, jobs each of which requires write access to regions that are to be accessed by no more than one of the processors "as a user writes an INSIGHT program 272 defining application tasks and the number of processor used for each task...Each configuration specification file defines a set of generic processors and the jobs partitioned among them" (col. 27, lines 56-62). This teaches associated task with each processor and only one task for that processor. Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify Hapner system to include the one task for one processor as taught by Somani to allow one processor to read or write the data to the database to keep the data consistency.

Regarding on claims 2 and 53, Hapner teaches the stored data includes items of the database comprises objects in an object database (col. 9, lines 1-3).

Regarding on claims 3 and 54, Hapner teaches the stored data includes data items that are provided as objects to an object-oriented application (col. 7, lines 25-30).

Regarding on claims 4 and 55, Hapner teaches an objected relational broker provides persistent storage of objects for an object-oriented application (col. 7, lines 34-35).

Regarding on claims 5 and 56, Hapner teaches the data is stored in a relational database with object-oriented extensions (col. 7, lines 1-10).

Regarding on claim 6, Hapner teaches the database comprises files that persistently store the data (col. 9, lines 12-13).

Regarding on claim 7, Hapner teaches the number of tasks accepted from the task sources is arbitrarily large (col. 9, lines 55-60).

Regarding on claim 8, Hapner teaches the number of task sources from which tasks are accepted is arbitrarily large (col. 17, lines 55-60).

Regarding on claim 9, Hapner teaches the regions are organized into contention spaces, the number of contention spaces being no less than the number of available processors (col. 9, lines 55-60).

Regarding on claim 10, Hapner teaches each of the jobs requires write access to data in no more than one of the contention spaces (col. 9, lines 55-60).

Regarding on claim 11, Hapner teaches the number of contention spaces is equal to the number of available processors (col. 9, lines 55-60).

Regarding on claim 12, Hapner teaches the organization of regions into contention spaces maximizes the throughput of the available processors in executing the jobs (col. 9, lines 55-60).

Regarding on claim 13, Hapner teaches the contention spaces are assigned dynamically to processors to maximize the throughput of the available processors (col. 9, lines 55-60).

Regarding on claim 14, Hapner teaches the tasks are accepted asynchronously (col. 9, lines 1-3).

Regarding on claim 15, Hapner teaches the tasks are accepted concurrently (col. 9, lines 55-60).

Regarding on claim 16, Hapner teaches the processors do not use shared memory (col. 18, lines 24-26).

Regarding on claim 17, Hapner teaches defining the jobs (task) for each task comprises defining hierarchy of subtasks in which the lowest level of the hierarchy contains the jobs (col. 18, lines 17-25).

Regarding on claim 18, Hapner teaches at least one of the tasks comprises a single job (col. 9, lines 55-60).

Regarding on claim 19, Hapner teaches a job generating a task to be performed (col. 9, lines 55-60).

Regarding on claim 20, Hapner teaches each of the tasks is completed with a certainty that is at least as high as the certainty that data updated in a requested database transaction is not lost once the transaction is committed (col. 9, lines 5-10).

Regarding on claim 21, Hapner teaches the region (database) comprises a single data item (data record) (col. 9, lines 20-30).

Regarding on claim 22, Hapner teaches the region comprises at least a million data items (data records) (col. 9, lines 20-30).

Regarding on claim 23, Hapner teaches the jobs are executed concurrently without having to wait for release of any write locks on any of the regions (col. 9, lines 55-60).

Regarding on claim 24, Hapner teaches more than one of the contention space is associated with one of the processors (col. 9, lines 55-60).

Regarding on claim 25, Hapner teaches each of the processors comprises a physical processor running at least one process (col. 9, lines 55-60).

Regarding on claim 26, Hapner teaches each of the tasks is generated by a user request (col. 9, lines 55-60).

Regarding on claim 27, Hapner teaches each of the contention spaces is associated with at least two processors one of which execute jobs and the other of which performs administrative functions with respect to the associated contention space (col. 9, lines 55-60).

Regarding on claim 32, Hapner teaches millions of regions belong to a contention space (col. 9, lines 55-60).

Regarding on claim 33, Hapner teaches additional jobs are created in connection with the execution of the jobs (col. 9, lines 55-60).

Regarding on claim 34, Hapner teaches jobs (task) are created by the additional jobs (col. 9, lines 55-60).

Regarding on claim 35, Hapner teaches the creation of the additional jobs is dependent on data read from the database (make file 72) in executing the jobs (col. 9, lines 55-60).

Regarding on claim 36, Hapner teaches additional jobs (sub-tasks) are created in connection with the execution of the jobs (overall task), and a process running on one of the processors execute the jobs and created the additional jobs, and in which at least some of the additional jobs are distributed among contention spaces served by other processor (col. 9, lines 55-60).

Regarding on claim 37, Hapner teaches the tasks relate to a commercial transaction (col. 11, lines 50-58).

Regarding on claim 38, Hapner teaches each of the jobs is assigned in index associated with the corresponding contention space (col. 9, lines 55-60).

Regarding on claim 39, Hapner teaches the indexes are used to load balance the jobs among processors (col. 9, lines 55-60).

Regarding on claim 40, Hapner teaches the database includes database units that are distributed among different physical location (col. 9, lines 27-32).

Regarding on claim 41, Hapner teaches each of the jobs comprises steps sub-tasks (col. 9, lines 27-32).

Regarding on claim 42, Hapner teaches execution of a job includes executing a portion of the steps, committing a database transaction representing those steps, and repeating until the job is completed (col. 9, lines 27-32).

Regarding on claim 43, Hapner teaches upon a failure to complete any portion of the steps, restarting the execution at the first step of the failed portion with at least the same level of certainty that the job will be completed as the certainty that data written in a requested transaction is not lost once the transaction is committed (col. 18, lines 35-41).

Regarding on claim 48, Hapner teaches synchronizing the executions of synchronization group of jobs to ensure correctness of results (col. 9, lines 27-32).

Regarding on claim 49, Hapner teaches the synchronizing includes assigning to each of the jobs of a synchronization group tag that identifies them as members of the group (col. 9, lines 27-32).

Regarding on claim 50, Hapner teaches the synchronizing includes assigning to each of the jobs of a synchronization group a quorum fraction representing the job's proportion of participation in the group (col. 9, lines 27-32).

Regarding on claim 51, Hapner teaches the jobs are not executed until all of the jobs in the synchronization group are ready for execution by a processor (col. 9, lines 27-32).

Claim 52 is the system claim recites similar limitations as to claim 1; therefore, claim 52 is rejected under the same reason as to claim 1.

11. Claims 57-61 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hapner et al. (US. Patent No. 5,727,203) in view of Zaiken et al. (US. Patent No. 5,907,848).

Regarding on claim 57, Hapner teaches a software object configured to be executed on a machine, the object comprising:

A job to be executed (threads), the job requiring access (write to the database 159) (col. 9, lines 25-30) to a region of a database that stores data persistently (persistence data portion 164) (col. 9, lines 55-60), the job including instruction and pointers to data in the region of the database (col. 9, lines 55-60);

Hapner teaches "as corresponding to multiples threads competing for the resource of both the database cache 160 and the persistence database portion 164" (col. 9, lines 55-60). Hapner does not explicitly teach an index that identifies a contention space of jobs that have competing requirement to write into the region of the database, the index distinguish the contention space from other contention spaces of jobs that do not have competing requirements to write into the region of the database. On the other hand, Zaiken teaches an index that identifies a contention space of jobs that have competing requirement to write into the region of the database, the index distinguish the contention space from other contention spaces of jobs that do not have competing requirements to write into the region of the database "as corresponding to if the file name in the records 20 matches a filename in the selected template 28, the transaction monitor program then searches for any existing index files 30 having job identifier data equal to the job identifier data in the record 20" (col. 11, lines 39-43). This teaches the index identifying the job in the index files to distinguish by comparing the job identifier in the file. Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify Hapner system to include comparing the job identifier in the index files of Zaiken on order to distinguish the jobs by comparing the job identifier in order to process the requested job.

Regarding on claim 58, Hapner teaches the stored data includes items of the database comprises objects in an object database (col. 9, lines 1-3).

Regarding on claim 59, Hapner teaches the stored data includes data items that are provided as objects to an object-oriented application (col. 7, lines 25-30).

Regarding on claim 60, Hapner teaches an objected relational broker provides persistent storage of objects for an object-oriented application (col. 7, lines 34-35).

Regarding on claim 61, Hapner teaches the data is stored in a relational database with object-oriented extensions (col. 7, lines 1-10).

12. Claims 68-83 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hapner et al. (US. Patent No. 5,727,203) in view of McClaughry et al. (US. Patent No. 5,933,825).

Regarding on claim 68, Hapner teaches a method comprising:

Maintaining a database that stores data persistently (col. 7, lines 15-16) and provides a primary level of guarantee that data written in a request transaction is not lost once the transaction is committed (col. 10, lines 11-22),

Accepting tasks from the task source for concurrent execution by multiple processors, at least some of the tasks having conflicting requirements to write into the same region of the database (col. 9, lines 54-65), and

Hapner teaches, "a mutex is created to corresponding to a piece of code, a portion data, some state, etc... when a thread has locked a mutex, it is said to "own" the locked mutex. In order for other threads to own the mutex, the first thread (i.e. the thread that locked the mutex) must unlock it. Thus mutexes provide a mechanism by which the programmer can control the serialization of multiple threads, ensuring that steps occur in a desired order and that the state corresponding to the mutex is maintained in a consistent manner" (col. 10, lines 11-22). This teaches the mutex lock is the guarantee that the thread holding the job will be executed and no data is lost. Hapner does not explicitly teaches providing a software mechanism that guarantees, as least to the primary level of guarantee, that the tasks will be executed without loss of data and without occurrence of any actual conflict with respect to the region of the database. Further more, McClaughry discloses providing a software mechanism that guarantees, as least to the primary level of guarantee, that the tasks will be executed without loss of data and without occurrence of any actual conflict with respect to the region of the database (There are also three types of write locks according to exemplary embodiments of the present invention. For example, in order to make changes to a file system object (e.g., update contents of a file), the application must acquire a Write Contents (WC) lock for that object. This can be accomplished

by evaluating the lock record (if any exists) associated with the object that is to be changed as will be described below with respect to FIG. 5. If a lock record exists for this object with a WC flag set to indicate an existing WC lock (i.e., another thread is currently writing this object), then this lock request is denied and the user can be notified. Otherwise, if no lock record currently exists for that object, or if one exists without a WC lock indication, then the WC field is set equal to a value indicating that a WC lock has been acquired, e.g., one. Unlike the RC lock, however, the WC field in the lock record may be a boolean field since only one write operation is permitted on an object at a time. When a WC lock is acquired each of the ancestors of that object acquires a Hierarchical Write Contents (HWC) lock which indicates that a write operation is taking place on an object lower in the hierarchy. The field associated with HWC locks in a lock record is also formatted for counting since it is permissible for a write operation in one sub-branch to co-exist with a write operation in another sub-branch of the hierarchy. For example, if a WC lock is acquired for both files F and G in FIG. 2, folder C would have two HWC locks indicated in its lock record. Analogously to read locks, the third type of write lock is the Write Children (WK) lock. In conjunction with the RK lock, the WK lock permits simultaneous writing of an object's contents and reading or writing of that object's children as will be described below with respect to FIG. 5.) (col. 6, lines 35-52). This read lock will resolve the conflicts and guarantee the write task to be executed without loss of data. Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify Hapner system to include write lock to resolve the conflict and to guarantee the write task to be

completed without losing the data as taught by Hapner in order guarantee the write task to be completed without any interruption.

Regarding on claim 69, Hapner teaches the stored data includes data items of the database that comprise object in an object database (col. 9, lines 1-3).

Regarding on claim 70, Hapner teaches the stored data includes data items that are provided as objects to an object-oriented application (col. 7, lines 25-30).

Regarding on claim 71, Hapner teaches an object relational broker provides persistent storage of objects-oriented application (col. 7, lines 34-35).

Regarding on claim 72, Hapner teaches the data is stored in a relational database with object-oriented extensions (col. 7, lines 1-10).

Regarding on claim 73, Hapner teaches sending to the task source acknowledgement of acceptance of the task (col. 9, lines 27-32).

Regarding on claim 74, Hapner teach sending to the task source a notification after completion of the accepted task (col. 9, lines 27-32).

Regarding on claim 75, Hapner teach the task is decomposed into jobs that are executed by different ones of the multiple processors in a manner that prevents any actual conflict between jobs a conflict upon complete execution of all tasks (col. 9, lines 27-32).

Regarding on claim 76, Hapner teaches the jobs are subject to a synchronization mechanism that enables a determination of the completion of a task (col. 9, lines 27-32).

Regarding on claim 77, Hapner teaches the synchronization mechanism includes a tag that identifies a job as participating in a group of jobs (col. 9, lines 55-60).

Regarding on claim 78, Hapner teaches the synchronization mechanism includes a quorum fraction that represents the job's proportion of participating in the group (col. 9, lines 55-60).

Regarding on claim 79, Hapner teaches determining whether the quorum fractions of all of the jobs of a group add to a complete quorum (col. 9, lines 55-60).

Regarding on claim 80, Hapner teach the task is notified of completion when all of the jobs have been completed (col. 9, lines 27-32).

Regarding on claim 81, Hapner teaches the task is assigned to a contention space (col. 9, lines 55-60).

Regarding on claim 82, Hapner teaches completion notification jobs are assigned for execution in the same contention space as the task (col. 9, lines 55-60).

Regarding on claim 83, Hapner teaches the database comprises an object-oriented database (col. 9, lines 1-3).

13. Claims 28-31 and 44-47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hapner et al. (US. Patent No. 5,727,203) in view of Somani et al. (US. Patent No. 5,524,212) and further in view of Murray (US. Patent No. 5,404,521).

Regarding on claim 28, both Hapner and Somani do not explicitly teach the distributing of the jobs includes maintaining a queuing system that has a capacity to receive jobs for execution at an arbitrarily large rate by adding processors proportionately to the number of jobs expected to require execution. However, Murray teaches the distributing of the jobs includes maintaining a queuing system that has a capacity to receive jobs for execution at an arbitrarily large rate by adding processors proportionately to the number of jobs expected to require execution "as corresponding to multiprocessor computer system having a plurality of processors (main processor and parallel processors) wherein an application program is separated into different tasks

which are executed in parallel by the system processor" (col. 3, lines 36-40). In addition, Murray also teaches, "if there are more than one tasks, the parallel processor will perform the execution for that task" (col. 3, lines 45-54). This teaches other preserved processor to execute the tasks when there are more than one task require for execution. Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include the queue system to call in for other preserved processors to execute task in the event there are more tasks waiting to execute as taught by Murray in order to enhance system performance in the multiprocessors system.

Regarding on claim 29, both Hapner and Somani do not explicitly teach the queuing system includes conceptual rows each of which can receive the jobs; however, Murray teaches the queuing system includes conceptual rows each of which can receive the jobs (col. 5, lines 10-15). Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include the queue system to include table having row and column to receives jobs as taught by Murray in order to enhance system performance in the multiprocessors system.

Regarding on claim 30, both Hapner and Somani do not explicitly teach each of the row is locked when jobs are being received in the row; however, Murray teaches each of the row is locked when jobs are being received in the row (col. 5, lines 23-26).

Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include row lock in the queue system as taught by Murray in order protect the row from being accessed to enhance system performance in the multiprocessors system.

Regarding on claim 31, Hapner teaches a job can be accepted by the corresponding processor for execution from any of the rows that is not locked (col. 10, lines 11-22).

Regarding on claim 44, Hapner teaches a row is locked only for reading when a processor is accepting jobs from that row (col. 10, lines 11-22).

Regarding on claim 45, both Hapner and Somani do not explicitly teach (1) the distributing of the jobs includes maintaining a queuing system that has a capacity to receive jobs for execution at any arbitrarily large rate, (2) the queuing system includes conceptual rows each of which can receive the jobs, and (3) the queuing system includes conceptual columns associated with respective contention spaces, the queuing system comprising a conceptual matrix of cells at the intersections of the rows and columns, in which each of the cells may be read from or written to without conflicting with reads and writes to other cells; however, Murray teaches (1) the distributing of the jobs includes maintaining a queuing system that has a capacity to receive jobs for execution at any arbitrarily large rate, (2) the queuing system includes conceptual rows

each of which can receive the jobs, and (3) the queuing system includes conceptual columns associated with respective contention spaces, the queuing system comprising a conceptual matrix of cells at the intersections of the rows and columns, in which each of the cells may be read from or written to without conflicting with reads and writes to other cells (col. 6, lines 30-59). Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include the queue system to call in for other preserved processors to execute task in the event there are more tasks waiting to execute as taught by Murray in order to enhance system performance in the multiprocessors system.

Regarding on claim 46, both Hapner and Somani do no explicitly teach the rows are associated with source of jobs and the number of rows is sufficient to permit all of the sources of jobs to load jobs into the queue concurrently; however, Murray teaches the rows are associated with source of jobs and the number of rows is sufficient to permit all of the sources of jobs to load jobs into the queue concurrently (col. 6, lines 15-21). Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include the queue system to load job currently when rows are available for process as taught by Murray in order to enhance system performance in the multiprocessors system.

Regarding on claim 47, both Hapner and Somani do no explicitly teach the number of rows is sufficient to permit jobs to be fetched for execution from all of the

columns concurrently; however, Murray teaches the number of rows is sufficient to permit jobs to be fetched for execution from all of the columns concurrently (col. 4, lines 59-61). Therefore, it would have been obvious to one ordinary skill in the art at the time of the invention was made to modify both Somani and Hapner to include the queue system to load job currently when rows are available for process as taught by Murray in order to enhance system performance in the multiprocessors system.

Conclusion

14. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Baoquoc N. To whose telephone number is at 571-272-4041, or unofficial fax number for the purpose of discussion (571) 273-4041 or via e-mail BaoquocN.To@uspto.gov. The examiner can normally be reached on Monday-Friday: 8:00 AM – 4:30 PM, EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached at 571-272-4107.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks
Washington, D.C. 20231.

The fax numbers for the organization where this application or proceeding is assigned are as follow:

(571) 273-8300 [Official Communication]

BQ To
April 13th, 2007